RAPII) PROCESS DESIGN
A SYSTEM ENGINEERING I) ESIGN METHOD

Ronald J. Aguilar
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91109-8099

ABSTRACT. A rapid method has been developed for the design of computing system capabilities and/or human work flow activities. This design method is call Rapid Process Design (RPD) and is based on the definition of processes and not requirements, This design originally was developed for use as a preliminary design tool supporting the Definition Phase for system development programs using the Rapid Development Methodology (RDM) [1]. The RPD method allows for quick design of automated computing processes for immediate computing system capability deliveries. A key clement of RPD is the definition of a process and its associated conditions and constraints. The process definition can be used to support evaluation and testing of a delivered computing system capability or human work flow.

IN'1'RODUCTION

Traditional methods of system engineering work well when there is time and adequate funds for system definition and requirements development. During traditional development formalized requirements can be passed to developers for system development, implementation and test. However, system engineering is being put under too much strain from limited time anti funds to support traditional methods of development. This is especially true for system engineering efforts on military programs. The RI'D method has been developed, and is being used at JPL, to rapidly define and design a system or series of processes for development. The processes are associated with a human performed set of tasks or computing system operations.

RPD can be developed during the Project Definition Phase of RDM. The RPD can provide definition and design for initial and future systems development without the burden of forma] req uirements. Forma] documentation can then be provided as system development proceeds, RPD can be used to shorten the Project Definition Phase. This is accomplished by obtaining support and process information from the developers and customer during this phase.

Traditional engineering generally supports major system deliveries with extensive formal testing of the system once implemented and operational. Since RDM supports multiple (incremental) deliveries, it is very costly to attempt formal testing for each RDM incremental delivery. RPD can be used to support evaluation and testing of each RDM incremental dc.livery. These incremental test efforts would not exclude formal testing. A formal test can be performed once the final RDM delivery has been made and the requirements have been completed and formalized.

RPD AND SYSTEM ENGINEERING

The RPD is mainly developed and completed during the Project Definition Phase of RDM. However, parts of the design such as the process description and process evaluation can begin in a prc project phase when the project is being defined and operational concepts are being formulated. From the prc project information the Mission Definition and the Operational Concept for human tasks or computer capabilities (system) can be developed.

Pre Project Phase. RPD can be used in the prc project phase when the mission definition and operational concept are loosely defined, to assist in clarifying the operational concept and capabilities that need to be designed and eventually developed.

Project Definition Phase. Project definition under RDM can be supported by RPD. The Mission Definition and Operational Concept should be provided in documents and completed during this phase of development. The information provided by these documents supports the completion of RPD by providing information in these areas:

(1) Mission Definition. The Mission Definition must express the mission objectives of the user tasks or capabilities that are being designed and developed and the mission impacts that may result given a set of known and perceived threats to the overall operations of the system. The Mission definition should supply information as follows,

(a) Mission objectives.

(b) Mission performance criteria.

(c) Mission reliability conditions.

(d) Mission impacts (security criteria).

(2) Operational Concept. The Operational Concept supports RPD by providing information for the process description and process evaluation (user task and present system capabilities). Operational concept information also supports process design and supplies information as follows:

(a) Task and capability (subject actions) descriptions.

(b) information flows (object to object),

(c) impacts to information flows.

In reference to RDM, the Project Definition Phase and development of the preliminary incremental delivery begin at the same time. The incremental deliveries continue until the system under development is complete. RPD can be used to design incremental RDM deliveries. Each design for a delivery could bc put in a single document to show the evolutionary developments as the deliveries of capabilities proceed.

The evolutionary changes can be:]) user tasks progressing towards computing system automation, or 2) new processes that could change how a user performs tasks or uses the completing system capabilities to complete tasks. The change could be in reference to process conditions such as an increase in performance, security or reliability.

RPD DEVELOPMENT

RPD development and documentation is based on three main sections. Each section provides information for process description, evaluation and design. The processes can be manual (performed by a human) or automated (performed by a computing system). The manual as well as computing system operations must be defined for future upgrade and/or automation.

Process Descriptions. The process description presents the activities of a user or computing system as task procedures external to the computing system, or as automated processes within the system. The user procedures are specified as part of the process design and can be automated in later project deliveries as the system evolves.

Process Evaluation. The process evaluation part of RPD includes a system process capabilities and constraints evaluation, This part of the RI'D is used to review and specify the capabilities that will be necessary to implement or integrate the functional portions of a process within the system or environment of the operational site, Hardware and/or software that is presently being used at the customer (operational) site may provide support for the system/processes being developed or integrated. The review also includes an evaluation of the present and future capabilities of the system being developed, and what capabilities must be added either by purchased or developed software. If development is necessary, then an evaluation of existing software development tools (i.e., compilers, linkers, report generators, etc.) will be provided and described as part of capabilities evaluation.

The process constraints and performance criteria must also be specified prior to the development of the process design. These are requirements (i.e., security, reliability, performance, etc.) that may be necessary for the completion of a functional part of a process. These requirements are used in the process condition description that corresponds to a specific process functional description. These Acquirements are general] y derived from government requirements or allocated functional requirements for the system or user operations.

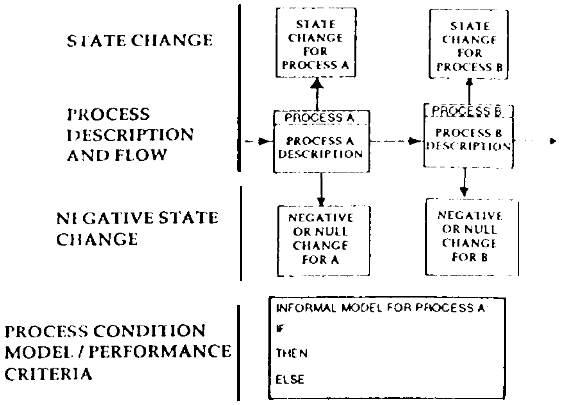
Process Design. The process design is the main part of tbc RPD and provides the definition information for each process of a user task or computing (automated) capability. This part of RPD uses a pictorial arrangement for the definition of a process and process flow that is described in more detail in the next section. The process design section includes process description, process state change conditions (positive and negative) and/or process conditions or criteria. This part of the RPD document can be divided into sections for the process design of each incremental de] ivery.

PROCESS DESIGN METHOD

The key element of RPD is the definition of a process. The process design must provide enough information for developers to automate/develop the process without formal requirements. The process (as designed) is composed of a process functional description, a state change condition, a negative state change condition and process conditions or criteria (See Figure 1). Sonic process conditions can be expressed as a model (i.e., process security conditions or constraints),

The design process is discussed using the terms Subjects and Objects. Subjects and objects can be used as part of a process condition model. Subjects arc considered: 1) users or administrators of the computing system, or 2) processes within the computing system that act on behalf of the user. Objects arc containers that hold information within the computing system (i.e., files, databases, messages, etc.). Subjects perform actions on Objects that cause State Change Conditions (i.e., open file for read or write access to the file information). A single action by a Subject is considered a process. A task (user) or capability (computing system) is comprised of several actions by a Subject.

Process Description and Flow. This part of the design provides the central description of each process that makes up an operation by a subject. The manual operations performed by a user (subject) arc important to specify. They may remain as user procedures or become candidates for future computer automation or work flow enhancements.



STATE CHANGE
PROCESS DESCRIPTION AND FLOW
NEGATIVE STATE CHANGE
PROCESS CONDITION MODEL / PERFORMANCE CRITERIA
STATE CHANGE FOR PROCESS A
STATE CHANGE FOR PROCESS B
PROCESS A
PROCESS A DESCRIPTION
PROCESS B
PROCESS B DESCRIPTION
NEGATIVE OR NULL CHANGE FOR A
NEGATIVE OR NULL CHANGE FOR B
INFORMAL MODEL FOR PROCESS A:
IF
THEN
ELSE

Figure 1. RPD Process Design Layout

The process functional description is presented as a step--by-step flow diagram.

State Change Condition. The State Change Condition has been added to the process design to show the changes to information duc to a subject's actions. Subjects act upon objects to cause State Changes (i.e., copy tile, send message, update database, change an access control list, update user password file, etc.). These changes can be either positive or negative depending on the actions of the subject, The positive changes are considered for the State Change Condition part of the design. The positive state change condition can be demonstrated as part of the correct operations of the process performed by a subject.

Negative State Change Condition. A negative condition occurs when a subject dots not perform the necessary process function or performs a process incorrectly. These types of state changes can be demonstrated during testing to be: 1) blocked from occurring by the computing system, 2) shown to occur infrequently and not considered a high risk, or 3) occurring with no means of prevention and known as a risk in the system.

Process Conditions. Process conditions (conditions imposed on the process) can include such things as performance criteria, reliability criteria and/or security constraints. A process condition, such as security constraints, can be expressed as an informal or formal model. Performance criteria can be expressed for each process based on a time element. The reliability condition can be expressed as the mean time to fail (MTTF) of the hardware components or software modules associated with the process. The security constraints model (as a model example) presents the conditions that are imposed on a functional part of a process. These conditions must be initiated before a process begins (performs some action on objects) or ends (completes some action on objects), The model not only assists in the development of a process but can also be used to assist in analysis and testing of the process. The model should correspond to the mission definition and impacts established for the overall system. Performance and reliability condition measurement criteria should be lcss than the overall mission definition for these conditions.

RPD AND SYSTEM DEVELOPMENT

The intent of RPD is to get developers involved in the design and development efforts as soon as possible during the early stages of RDM. Developers can address user tasks and assist in making decisions about what can be automated and what subject actions need to be changed. The developer can participate in process definition, process evaluation and process design. In terms of process development, the developer can use each part of the process design.

Process **Description and Flow.** This information is used to provide computer software program development and program flow. The information is also used to define the user actions as well as what actions can be programmed into a computing system or need to be redefined.

State Change Condition. This information defines for the developer the outcome of the process.

Negative State Change Condition. A negative condition expresses the types of conditions that should not occur given a user or computer action. Using this information a user task could be changed. in reference to a computing system, alarms, corrective software or software checking mechanisms could be added to a program to mitigate or prevent a negative state change condition.

Process Condition Model. A simple mode] can be used to express a condition or constraint that is imposed on the process. The model can express to developers how and where the condition or constraint is to be implemented. Model information can also assist in interpreting the condition or constraint (requirement) in terms of implement at ion.

Performance and/or Reliability Criteria. This criteria provides information as to what user tasks and capabilities should change for bcttcr process flow.

RPD ANI) SYSTEM EVAI,UATION

Evaluations of incremental deliveries must bc providcd during dcvclopmcnt and once the work flow upgrade or computing capability is operational at the customer's site. The evaluation during incremental development can be used to assess the progress of the developers to ensure that RDM schedules will be met. An evaluation, in the form of an operation] test, must also be performed once a delivery has been completed. The results of this test can be used for future delivery changes and customer support.

The RDM incremental delivery schedule allows partial system implementation to be rapidly made (within a matter of months). There is little time however, for evaluation of the development by management to ensure that development will be done according to design and schedule. Project managers have the ability, using the process design, to evaluate the progress of the system development effort. This can be done by performing a compliance evaluation (compare development to the process design) during development to determine how well developers are meeting the design criteria. In areas of uncertainty, rapid prototyping may be used to evaluate a process for capability, performance, security, reliability, etc. The comparisons arc made based on the Mission Definitions for the overall system anti the performance, security and reliability conditions associated with each process.

RPD provides the support for Operational Testing of a delivery. This is an evaluation to ensure that the customer needs arc being met. The Operational Test may include:

(1) Process Flows used as test procedures.

(2) State Change Conditions that provide the test pass criteria.

(3; Negative State Change Conditions that provide the test fail criteria.

(4) constraint or Condition Models to evaluate how well the constraints and conditions arc being provided by the developed process. The evaluation can include analysis, test demonstrations, stress tests and penetrations tests (security evaluations).

(5) Performance and reliability conditions can be tested using the criteria provided by the Mission Definition and the process design. The process performance and reliability measured values should be within the criteria limits defined in the Mission Definition.

CONCLUSIONS

Traditional design methods using formal require.ment cannot be used in a timely and cost effective manner to support RDM. The RPD method can provide a rapid design of processes in support of programs using RDM. RPD accomplishes this in several ways:

(1) Provides a flexible design which uses process definition and not formal requirements.

(2) Involves developers and customers in the design process.

(3) Provides a means of using process description and process evaluation to evaluate the present state of the customer job-related activities and associated computing systems.

(4) Adds conditions such as performance, reliability and security to assist developers in the interpretation of conditions necessary for process development.

(5) Provides a means to evaluate development progress and adequacy of the delivered system in terms of customer needs.

There are several attributes, as expressed in this paper, for the design and development of systems that are provided by using RPD. However, there are some aspects of system engineering and RPD use that need further review and analysis. These include:

(1) RPD support for the formalization of requirements.

(2) RPD use in formal testing.

(3) Engineering change process and changes to RPD.

(4) Use of modeling tools in support of process design and evaluation.

REFERENCES

[1] Riley, Norman B., and William H. Spuck, "System Engineering and the Rapid Development Method," Proceedings, Fourth Annual International Symposium, National Council on Systems Engineering (NCOSE), 1994.

AUTHOR'S BIOGRAPHY

Ronald J. Aguilar is a System/Security Engineer working for Telos Corporation at NASA's Jet Propulsion Laboratory, where he provides system and security engine.ering support for military C^{3}1 systems. In the area of security engine.ering he provides the policy, requirements, design and security evaluations for computing and network systems. His technical background extends over 13 years and encompasses system engineering and development of multiprocessing anti multitasking computing systems, communication interface devices and secure communication protocols. Mr. Aguilar was codeveloper of the Distributed System Analysis Method and Mission Based Risk Analysis Process for secure system certification evaluation. He has also provided J PL with a methodology for security evaluation of military C^{3}I systems.